

Tietokantasovelluksen harjoitustyö

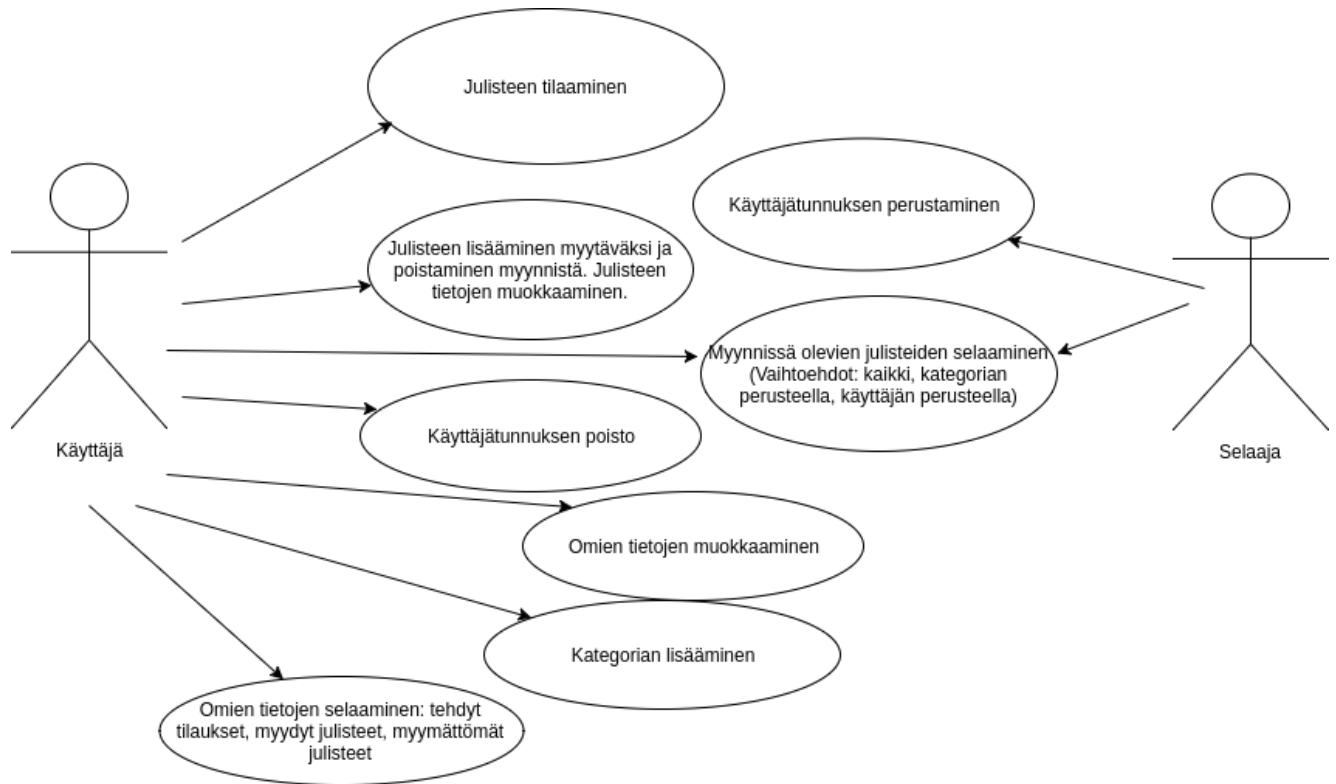
1. Johdanto

Tietokantasovelluksen harjoitustyön aiheena on uniikkeja julisteita myyvä verkkokauppa. Verkkokauppaan voi perustaa oman käyttäjätunnuksen, jolla pääsee lisäämään myytäviä tuotteita sekä tilaamaan eli ostamaan julisteita. Tämän järjestelmän on tarkoitus helpottaa uniikkien julisteiden ostamista ja myymistä verkossa.

Työ toteutetaan Helsingin yliopiston tietojenkäsittelytieteen laitoksen users-palvelimella. Web-sovelluksen alustajärjestelmän täytyy tukea PHP-ohjelmointikieltä. Selaimella ei tarvitse olla erityisominaisuuksia tuettavien ohjelmointikielten suhteen. Ohjelmisto edellyttää PostgreSQL-tietokannan käyttöä.

2. Yleiskuva järjestelmästä

Käyttötapauskaavio



Käyttäjärühmät

Selaaja: Kuka tahansa sivustolla vieraileva käyttäjä.

Käyttäjä: Järjestelmään rekisteröitynyt käyttäjä.

Käyttötapauskuvaukset

Selaajan käyttötapaukset:

- Selaaja voi perustaa käyttäjätunnuksen ja ryhtyä järjestelmän käyttäjäksi. Käyttäjätunnuksen perustamiseen tarvitaan etunimi, sukunimi, syntymäaika, osoite, puhelinnumero ja sähköpostiosoite.
- Etusivulla näkyy 10 parasta myyjää listattuna eniten julisteita myyneestä myyjästä alkaen.

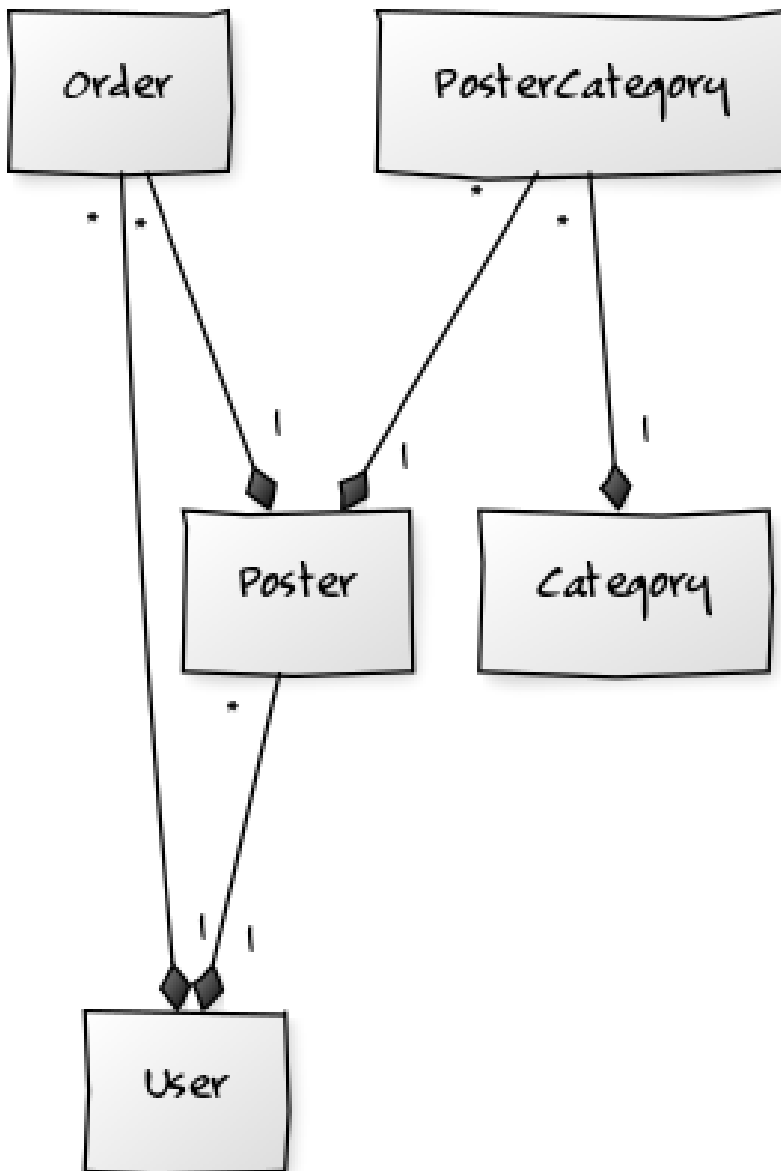
- Selaaja voi selata käyttäjiä, kategorioita ja sivustolla myynnissä olevia julisteita edellä mainittujen mukaan listattuna
 - Posters-sivulla näkyy kaikki myynnissä olevat julisteet
 - Categories-sivulla on listattuna kaikki tietokantaan lisätyt kategoriat linkkeinä. Linkistä pääsee katsomaan kaikkia kyseiseen kategoriaan liittyviä julisteita.
 - Users-sivulla näkyy kaikki rekisteröityneet käyttäjät linkkeinä. Linkistä pääsee katsomaan käyttäjän myynnissä olevia julisteita.

Käyttäjien käyttötapaukset:

- Käyttäjä voi sisäänkirjautuneena toteuttaa kaikkia selaajan käyttötapauksia.
- Omien tietojen muokkaaminen: Käyttäjä voi muokata kaikkia tietojaan My account -sivun kautta.
- Käyttäjä voi julisteita selatessaan valita julisteen ja lisätä sen ostoskoriin.
- Käyttäjä voi poistaa julisteita ostoskorista.
- Käyttäjä voi tehdä ostoskorin kautta tilauksen haluamistaan julisteista.
- Käyttäjä voi lisätä myyntiin julisteitaan ja poistaa niitä halutessaan myynnistä. Käyttäjän tulee asettaa julisteelle nimi, artistinimi, korkeus, leveys, hinta ja kuva. Hinnan, korkeuden ja leveyden tulee olla kokonaisluku. Kuvan tulee olla tiedostomuotoa: .png, .jpg, .jpeg, .gif tai .bmp.
- Julisteelle voi asettaa yhden tai useamman kategorian. Yhdellä julisteella voi olla monta kategoriaa ja kategoria voi viitata moneen julisteeseen.
- Käyttäjä voi lisätä uuden kategorian.
- Käyttäjä voi selata My account -sivulta omia myymättömiä julisteitaan.
- My account -sivulta pääsee My orders -linkin kautta selaamaan käyttäjän tekemiä tilauksia.
- My account -sivulta pääsee My sales -linkin kautta selaamaan käyttäjän myymiä julisteita.

3. Järjestelmän tietosisältö

Käsitekaavio



Tietokohde: Username		
Attribuutti	Arvojoukko	Kuvailu
id	Kokonaisluku	Tunniste, uniikki
firstName	Merkkijono, max. 50 merkkiä	Etunimi
lastName	Merkkijono, max. 50 merkkiä	Sukunimi
address	Merkkijono, max. 50 merkkiä	Asunnon katuosoite
postalcode	Merkkijono, max. 5 merkkiä	Postinumero
city	Merkkijono, max. 50 merkkiä	Kaupunki
name	Merkkijono, max. 20 merkkiä	Käyttäjänimi
password	Merkkijono, max. 50 merkkiä	Salasana
email	Merkkijono, max 500 merkkiä	Sähköpostiosoite

Tietokohde: Poster		
Attribuutti	Arvojoukko	Kuvailu
id	Kokonaisluku	Tunniste, uniikki
name	Merkkijono, max. 50 merkkiä	Julisteen nimi
publisher	Merkkijono, max. 50 merkkiä	Julkaisija, viittaa Username taulun pääavaimeen
artist	Merkkijono, max. 50 merkkiä	Julisteen tekijän artistinimi
price	Kokonaisluku	hinta
location	Merkkijono, max. 50 merkkiä	Sijainti, kaupunki
height	Kokonaisluku	Julisteen korkeus
width	Kokonaisluku	Julisteen leveys
image	Merkkijono, rajoittamaton koko	Kuva
filetype	Merkkijono, max 50 merkkiä	Kuvan tyyppi

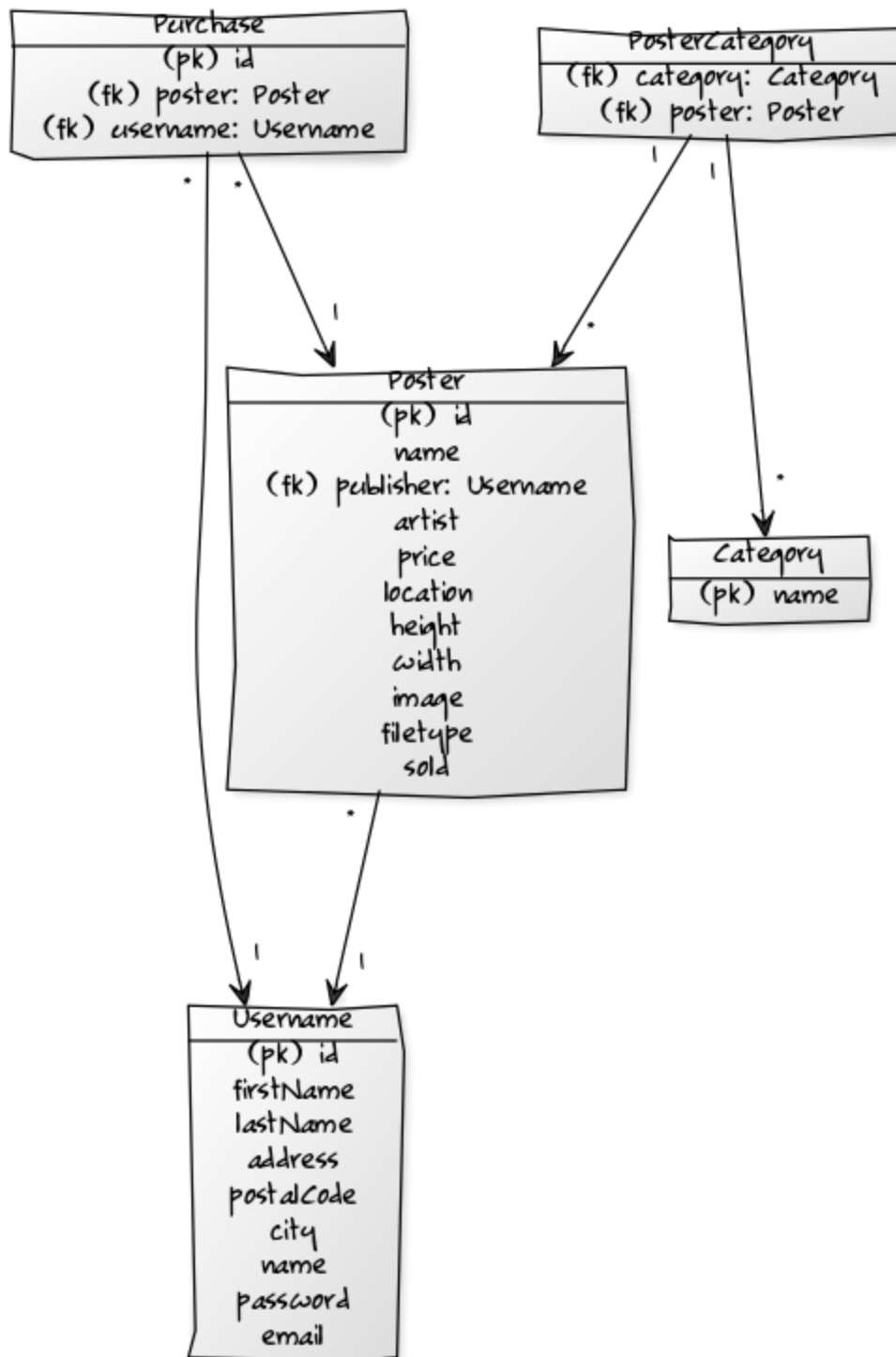
sold	Totuusarvomuuttuja	Julisteen myynnin tila, Totuusarvo TRUE (myyty) tai FALSE (myynnissä), alussa automaattisesti FALSE
------	--------------------	---

Tietokohde: Category		
Attribuutti	Arvojoukko	Kuvailu
name	Merkkijono, max. 20 merkkiä	Kategorian nimi

Tietokohde: PosterCategory		
Attribuutti	Arvojoukko	Kuvailu
category	Merkkijono, max. 20 merkkiä	Kategorian nimi, viittaa Category taulun pääavaimeen
poster	Kokonaisluku	Julisteen tunniste, viittaa Poster taulun pääavaimeen

Tietokohde: Purchase		
Attribuutti	Arvojoukko	Kuvailu
id	Kokonaisluku	Tunniste, uniikki
poster	Kokonaisluku	Julisteen tunniste, viittaa Poster taulun pääavaimeen
username	Kokonaisluku	Käyttäjän tunniste, viittaa Username taulun pääavaimeen

4. Relaatiotietokantakaavio



5. Järjestelmän yleisrakenne

Tietokantasovellusta tehdessä on noudatettu MVC-mallia. Kontrollerit, näkymät ja mallit sijaitsevat hakemistoissa controllers, views- ja models. Views kansion näkymät ovat jaettu alakansioihin poster, user ja category sen mukaan mihin ne liittyvät. Tietokantataulujen luontiin liittyvät tiedostot ovat sijoitettu sql kansioon. Käytetyt apukirjastot ovat sijoitettu hakemistoon lib ja asetukset ovat tiedostossa settings.php.

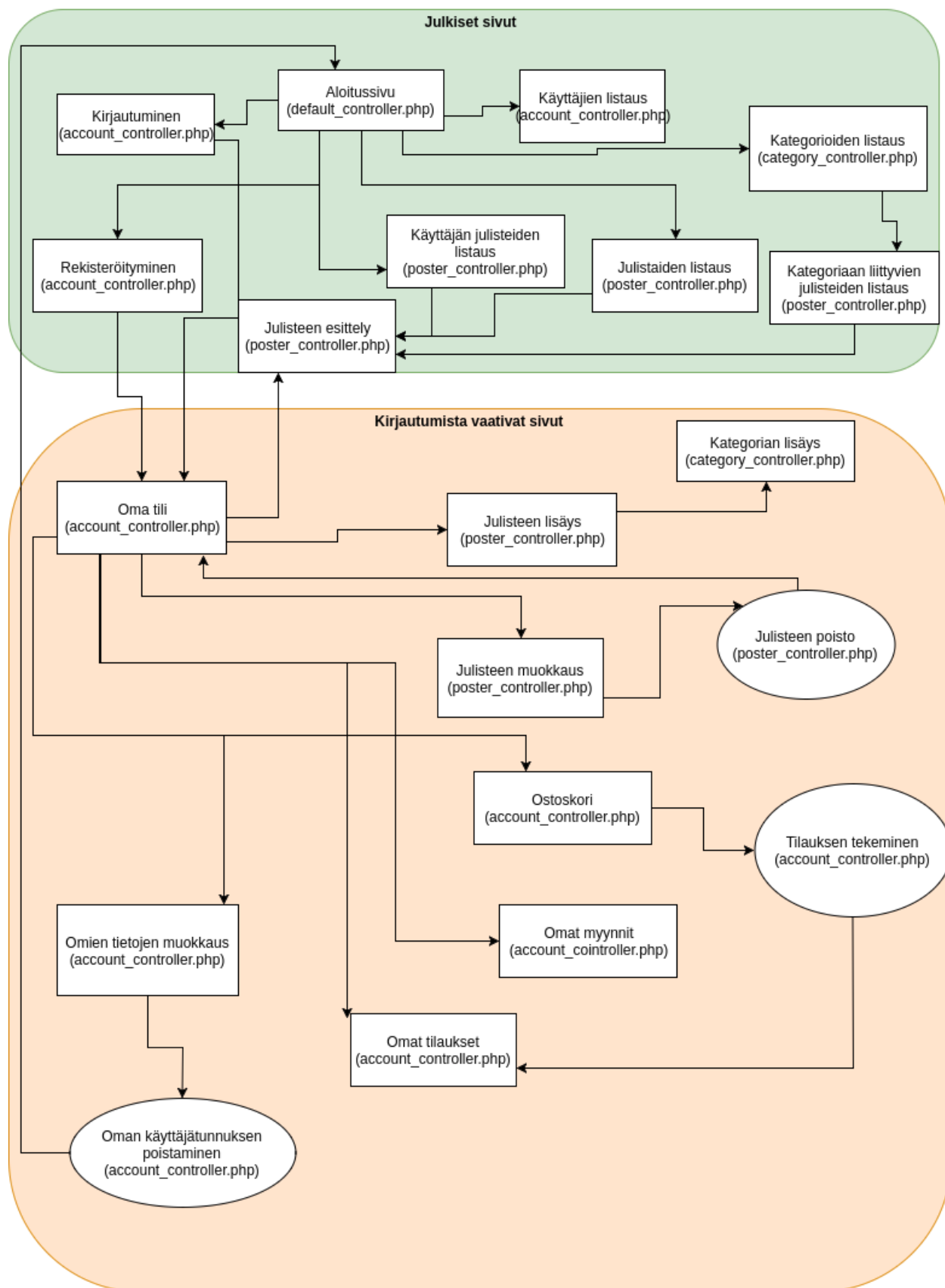
Javascriptiä on käytetty poistotoiminnon yhteydessä varmistamaan haluaako käyttäjä poistaa oikeasti kohteen, scripti löytyy site.js-tiedostosta, joka on sijoitettu kansioon assets/js.

Kaikki tiedostonimet ovat kirjoitettu pienellä.

6. Käyttöliittymä ja järjestelmän komponentit

Käyttöliittymäkaaviossa on esitetty tietokantasovelluksen käyttöliittymän näkymät ja niiden olennaiset yhteydet. Tietokantasovelluksessa on lisäksi näkyvillä aina navigaatiopalkki, josta pääsee aina julisteiden listaus, käyttäjien listaus, kategorioiden listaus, rekisteröitymis-, kirjautumis- ja etusivulle. Lisäksi käyttäjän ollessa kirjautuneena sisään, hän näkee navigaatiopalkissaan kirjautumis- ja rekisteröitymissivun tilalla oma tili ja ostoskorisivun.

Käyttöliittymäkaavio



7. Asennustiedot

Asenna sovellus kopioimalla sen tiedostot palvelimen nettiin näkyvään hakemistoon (esim. usersin htdocs-hakemisto). Aseta sen jälkeen tietokannan yhteystiedot oikeaksi tiedostoon config/database.php. Tällä hetkellä ohjelmisto edellyttää PostgreSQL-tietokannan käyttöä, ja sen asennustiedot ovat config/database.php tiedostossa valmiina.

8. Käynnistys- / käyttöohje

Ohjelman WWW-sovellus on helppokäyttöinen ja sisältää riittävästi informaatiota käyttöä varten.

Nettisivut:

<http://elisanur.users.cs.helsinki.fi/postershop/>

Testaamista varten toteutettu esimerkkikäyttäjä:

Käyttäjätunnus: makemies

Salasana: makeonjees

9. Testaus, tunnetut bugit ja puutteet & jatkokehitysideat

Olen testannut ohjelmaa toteuttamalla itse testejä mekaanisesti nettisivuilla, suorittamalla tietokannan hakutoimintoihin tarkoitettuja SQL-lausekkeita users palvelimella ja käyttämällä Kint-apuohjelman dump metodia. URL-osotteita on testattu antamalla erilaisia syötteitä parametreiksi. Nappien toimintaa on testattu käytännössä. Lomakkeita on testattu useilla erilaisilla syötteillä. Testejä on toteutettu myös selaimen inspect element -toiminnolla.

Validointi on myös osa testausta, ja ohjelmaan on toteutettu erilaisia validointi-metodeja, jotka testaavat käyttäjän syötteitä automaattisesti. Validointi-metodit testaavat käyttäjän syötteitä aina kun tietokantaa yritetään muokata. Merkkijonoista on tarkastettu esimerkiksi määrä, onko merkit aakkosia, numeroita tai tiettyjä erikoismerkkejä. Sähköpostiosoitteet ovat validoitu PHP:n valmiilla validointimetodilla, joka tarkastaa käytännössä vaikuttaako osoite sähköpostiosoitteelta. Kuvan lisäystä on testattu PHP:n omalla metodilla, joka tarkastaa onko

tiedosto kuva. Lisäksi kuvan kokoa on rajoitettu kahteen megatavuun, vaikka palvelimen puolesta kuvan koko voisi olla kahdeksan megatavua.

Ohjelmassa on tunnettuja bugeja tai puutteita. Koska harjoitustyöhön varattu aika loppui osaltani hieman kesken, jäi esimerkiksi koodin perusteellinen refaktoroiminen tekemättä. Koodissa on valitettavasti toistuvia asioita, jotka olisi voinut toteuttaa paremmin. Tiedostorakenne on myös karkeahko, ja sitä olisi voinut hieman parantaa. Julisteita käsittelevät toiminnot ovat poster_controllerissa, mutta esimerkiksi modeleissa löytyy hieman epäjohdonmukaisuutta. Ohjelmasta puuttuu sähköpostin verifikaatio. Tällä hetkellä on oletuksena että käyttäjät ovat Suomessa asuvia, ja laittavat yhteystietoihinsa osoitteen, joka on Suomessa. Postinumeroa ei ole kuitenkaan validoitu tarkemmin. Ohjelmaan voisi lisätä tiedoston Suomen postinumeroista, ja verrata postinumeroa niihin. Osoitteen validointia voisi muutenkin parantaa.

Julisteen lisäyksessä oli paljon vaikeuksia kuvan lisäämisen vuoksi. Koska kuvan lisääminen on toteutettu samalle lomakkeelle, tuli usein vastaan tilanteita, joissa \$_POST muuttuja tyhjenee kokonaan kuvan lisäämisen epäonnistuessa. Ongelmaa ei saatu kunnolla korjattua, mutta siitä on varoitettu nettisivulla. Kuvan lisäämiselle olisi ollut hyvä toteuttaa kokonaan oma näkymä, jolloin ongelmalta olisi välttytty.

Ohjelmaa voisi jatkokehittää paljon. Ohjelmaan voisi toteuttaa esimerkiksi erilaisia maksu- tai toimitustapoja. Julisteen myyjä voisi valita omista tiedoistaan minkälaisia maksu- tai toimitustapoja hän itse haluaa toteuttaa. Esimerkkinä maksu- tai toimitustavoista voisi olla verkkopankkimaksu, jonka jälkeen toimitus postituksena tai esimerkiksi toimitus henkilökohtaisesti valitulla alueella, ja käteismaksu paikan päällä. Verkkokauppaan voisi toteuttaa myös vapaavalintaisen kehyksen valinnan ja oston, minkä voisi toteuttaa monella eri tapaa. Ohjelmaa voisi laajentaa niin että verkkokauppa olisi toimiva kaikissa maissa ja toimisi maakohtaisesti alussa valitun maan mukaisesti.

10. Omat kokemukset

Tämä ohjelma oli ensimmäinen tekemäni tietokantasovellus, sekä ensimmäinen web-palvelinohjelmointia vaativa oma projekti. Projekti oli haastava ja työläs, sekä odotukset työmäärästä huomattavasti todellisuutta pienemmät. Projekti kesti noin 7 viikkoa ja keskimääräinen työmäärä viikossa oli noin 25h. Tietokantasovelluksen harjoitustyöhön varattu aika ei mielestäni riittänyt valitsemani aiheen hyvään ja perusteelliseen toteuttamiseen. Tämä johtui suurimmaksi osaksi siitä, että sovelluksen toteuttamista varten joutui opiskelemaan niin paljon uusia asioita.

Käytin sovelluksen toteuttamiseen NetBeans ohjelmaa, joka on integroitu ohjelmointiympäristö, ja ohjelmointikielenä oli PHP. En ollut koskaan aiemmin käyttänyt PHP-ohjelmointikieltä, ja alussa vei paljon aikaa tutustua kieleen sekä opetella yksinkertaistenkin asioiden toteuttamista. PHP:ssa käytetty sivupohjamoottori Twig aiheutti myös usein päänvaivaa. NetBeans ei tunnistanut HTML tiedostoissa Twig:n kieltä siten, että se olisi osannut antaa korjausehdotuksia tai toteuttaa esimerkiksi automaattista täydennystä. HTML tiedostojen muokkaus oli myös hankalaa, koska Twig-kieltä ei pystynyt kommentoimaan pois, joten testaus piti toteuttaa aina lisäämällä ja poistamalla koodia.

MVC-mallin hahmottaminen ohjelmointiympäristössä oli aluksi hankalaa. Sovelluksen edetessä tiedostomäärät ja koodirivit kasvoivat, mikä edelleen vaikeutti kokonaisuuksien hahmottamista. Välillä MVC-mallin toteuttaminen käytännössä saattoi myös hieman unohtua, ja ajoittain tiedostorakenteen pitäminen siistinä oli vaikeaa. Oman tietokantasovelluksen perustoimintojen teko oli kohtalaisen helppoa sen jälkeen, kun oli hahmottanut MVC-mallin myös käytännön tasolla. Ohjelman edetessä huomasi jatkuvasti oppineensa uusia asioita ja perusasioiden helpottuvan. Opinkin mielestäni projektin aikana valtavasti uusia asioita. Kokonaisuudessaan tietokantasovelluksen harjoitustyön tekeminen oli erittäin antoisaa.